SSRG International Journal of Computer Science and Engineering
ISSN: 2348–8387 / https://doi.org/10.14445/23488387/IJCSE-V11I6P105

Volume 11 Issue 6, 30-38, June 2024
© 2024 Seventh Sense Research Group®

Original Article

Enhanced Plant Leaf Disease Identification by Integrating 2DCNN and Transfer Learning for a Content-Based Image Retrieval System

M. Selvarani[1], M. Arulselvi[2]

[1,2]Department of Computer Science & Engineering, Annamalai University, Tamilnadu, India.

[1]Corresponding Author : selvarani.murugaiyan@gmail.com

Received: 21 April 2024 Revised: 26 May 2024 Accepted: 09 June 2024 Published: 29 June 2024

Abstract - *Efficiently managing plant leaf conditions in large agricultural areas requires automated detection of leaf diseases. This research presents an integrated approach utilizing Content-Based Image Retrieval (CBIR) systems for the automatic identification of plant leaf diseases. The CBIR system employs preprocessing, segmentation using the graph cut superpixel method and feature extraction via 2D Convolutional Neural Network (CNN) models, which is focused primarily on detecting colorful plant flake complaints prevalent in blueberry, cherry and apple crops. Additionally, Gradient Boosting Trees (GBT) serve as classifiers to point to database stores of class predictions made by the classifiers for plant leaf images. Comparable deep learning models such as InceptionV3 and ResNet50 in the proposed CBIR system extract high-level features from images, predict leaf classes and store them in the database. Euclidean similarity measures between feature representations further assess the CBIR system's overall performance, with deep learning models successfully supporting retrieval and image classification tasks when compared with 2D CNN models.*

Keywords - *Content-based image retrieval, 2DCNN, GBT, Inception V3, Euclidean similarity measure, Gradient boost, ResNet50.*

1. Introduction

Agricultural production stands as one of humanity's oldest methods of sustenance by providing a crucial source of income for populations worldwide. Plants play a vital role not only for humans but also for animals which serves as a source of food, oxygen and various other necessities. When a plant yields to disease, the repercussions are felt throughout the ecosystem by impacting all living organisms in some capacity. This reality underscores the interconnectedness of life forms and emphasizes the importance of proactive measures to safeguard agricultural productivity and ecological balance [1]. Each year, the global population grows by roughly 1.6%, leading to a heightened demand for plant-based products.

Consequently, safeguarding crops against diseases becomes increasingly vital in meeting the escalating need for food quantity and quality [2]. Automated plant disease identification using leaf images marks a significant breakthrough in agriculture. This advancement relies on cutting-edge technologies like machine learning and computer vision to swiftly and accurately detect diseases based on visual cues present on plant leaves [3]. Farmers face challenges in predicting diseases promptly due to limited knowledge and vast cultivation areas. However, the emergence of Machine Learning (ML) and its Deep Learning (DL) has revolutionized conventional disease prediction methods, significantly enhancing accuracy levels [4].

Reducing impulse noise in images is crucial for preserving image integrity and enhancing visual quality, with techniques such as median filtering, adaptive filtering, edge-preserving filters, wavelet denoising and machine learning-based methods being commonly employed for this purpose in image processing and computer vision analysis [5]. Superpixels are homogeneous and locally coherent structures, oversample information or scale resolutions in images with numerous methods available for image segmentation or super pixelization. Each method utilises various techniques tailored to address factors like light and shadow that can impact image acquisition. Within image processing research, several algorithms have been explored for generating superpixels, some lacking the ability to effectively control segment size, number and compactness, with these algorithms broadly categorized into graph-based methods and gradient-ascent-based approaches [6]. Deep Learning Convolution Neural Network (DLCNN) can be

This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

used to detect and classify tomato plant leaf diseases. It consists of many convolution layers, batch normalization, activation, max-pooling, fully connect, SoftMax and classification. The network architecture consists of three blocks, the first one includes convolution, batch normalization, activation function and max-pooling. The remaining blocks include convolution, activation function, and max-pooling, followed by a fully connected layer, SoftMax layer and classification layer [7]. Dense Net is presented in the paper, which is compared with the previous image recognition network, it solves the problem of gradient disappearance of deep network, strengthens the propagation of features, encourages feature reuse and reduces model parameters. It is characterized in that each layer in the network is directly connected to its front layer to achieve feature reuse. At the same time, each layer of the network is designed to be particularly narrow, i.e., only very few feature maps are learned to reduce redundancy [8].

The Deep Belief Network (DBN) method in deep learning is employed for feature extraction and classification, which represents a burgeoning research domain due to the proliferation of large datasets [9]. CBIR with deep learning autonomously determines similarity across shape, color and texture attributes by using various architectures fine-tuned with max-pool overlapping pooling. Models like VGG-16, VGG-19, Xception, InceptionResNetV2, DenseNet201, MobileNetV2 and NAS Net Large are assessed for optimal performance, first through classification and then through CBIR feature extraction [10]. Rapid access to large image databases and efficient extraction of identical images from a query image pose significant challenges in CBIR systems. Efficiency relies heavily on calculating feature representation and similarity. To address this, a basic yet powerful deep learning system centered on Convolutional Neural Networks (CNN) is presented, comprising feature extraction and classification for swift image retrieval [11]. Recently, the gradient boosting algorithm family has expanded with notable proposals such as XGBoost, Light GBM and CAT Boost, emphasizing both speed and accuracy. XGBoost is a scalable ensemble technique known for its reliability and efficiency in solving machine-learning challenges. Light GBM prioritizes extremely fast training performance by selectively sampling high-gradient instances. CAT Boost enhances model accuracy by modifying gradient computation to prevent prediction shifts [12]. The authors utilized Decision Tree, Random Forest, Gradient Boosting and Artificial Neural Network algorithms to analyze datasets, with Gradient Boosting demonstrating superior performance. Specifically, Gradient Boosting achieved an accuracy rate of 86.67% in the dataset. These findings indicate that the Gradient-boosting-based classification technique can effectively support prediction research [13].

In the context of plant disease identification, CNNs have shown great promise. For instance, Mohanty et al. [14] employed deep CNNs to identify 26 different diseases in 14 crop species, achieving a high accuracy rate. Similarly, Sladojevic et al. [15] used a deep learning model to identify 13 different types of plant diseases from leaf images. Several studies have explored CBIR for plant disease identification. Brahimi et al. [16] developed a CBIR system using CNN features for plant disease retrieval. Similarly, Picon et al. [17] utilized a CBIR approach with deep learning features to classify and retrieve diseased leaf images, demonstrating the effectiveness of combining CBIR with deep learning techniques.

Two prominent approaches in the realm of CBIR for plant leaf disease identification have developed: one employing 2DCNN and the other utilizing Transfer Learning, specifically through architectures like Inception V3 and ResNet50. The primary objective of this study is to provide a comparative analysis of these two CBIR systems, elucidating their methodologies, strengths, weaknesses and performance metrics. By examining and contrasting these approaches, it is aimed to discern which method offers superior efficacy and applicability in the context of plant leaf disease identification. The first CBIR system utilizes 2DCNN for feature extraction, followed by a series of preprocessing steps, including noise removal and segmentation, using the superpixel graph cut method.

Contrastingly, the second CBIR system leverages Transfer Learning, specifically adopting pre-trained architectures like Inception V3 and ResNet50. By transferring knowledge from models trained on large-scale datasets, this approach obviates the need for extensive data and computational resources by making it highly efficient. The system further refines features using similar preprocessing techniques and employs classifiers such as GBT. Through this comparative analysis aims to provide insights into the efficacy of these methodologies in real-world scenarios of plant leaf disease identification. By scrutinizing their performance metrics, computational efficiency and adaptability to varying datasets, the study informs researchers and practitioners about the most effective approaches in the domain of CBIR for agricultural applications.

Overall, this study serves as a comprehensive exploration of CBIR systems for plant leaf disease identification, shedding light on the advancements, challenges and future directions in this burgeoning field of research. The work has been organized as follows: Section 2 covers with methodology, Section 3 explores the result analysis and Section 4 deals with the conclusion of the study.

2. Methodology

The methodology outlined in this study outlines the systematic approach employed to compare two prominent

CBIR systems for plant leaf disease identification. These systems, based on 2DCNN and Transfer Learning, respectively, offer distinct approaches to feature extraction, classification and performance evaluation. This methodology encompasses several key steps, including data collection and preprocessing, segmentation, feature extraction, refinement, classification, similarity measure and performance evaluation. Each step is meticulously designed to ensure the robustness, accuracy and efficiency of the CBIR systems in identifying plant leaf diseases. The overall flow diagram of the study has been depicted in Figure 1.

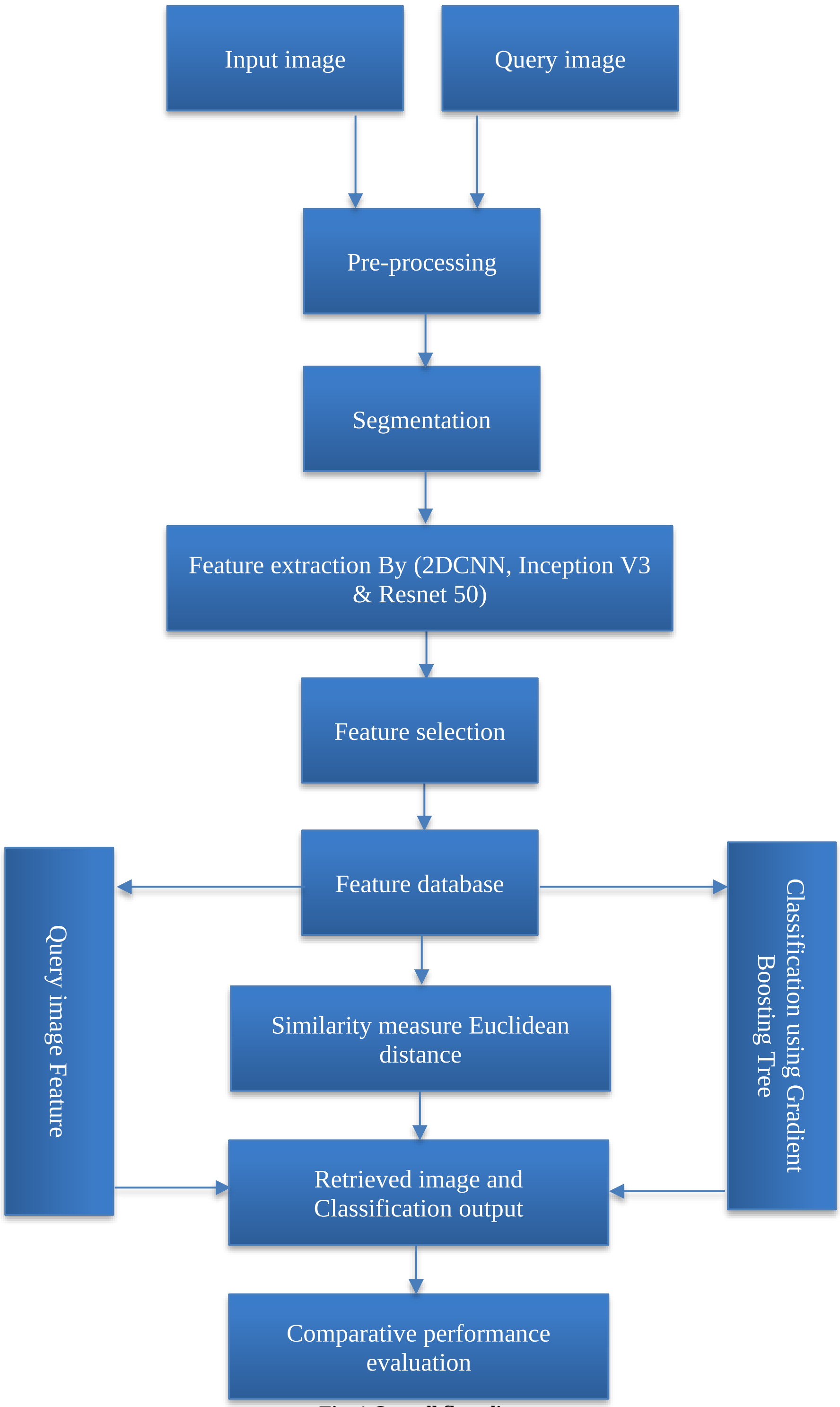

Fig. 1 Overall flow diagram

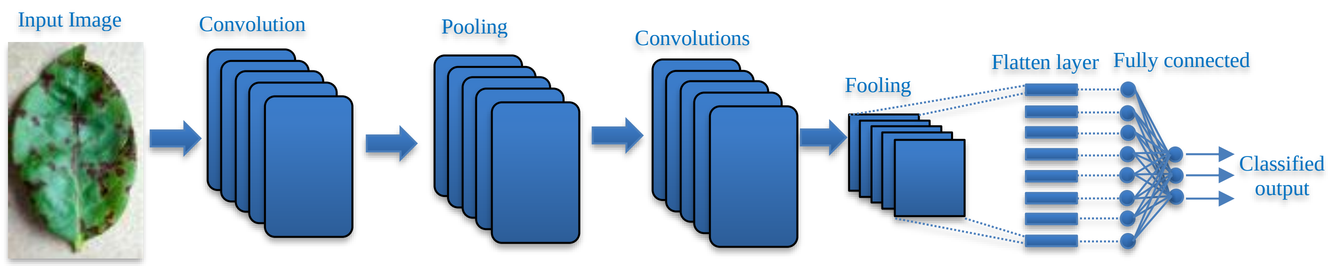

Fig. 2 Architecture diagram of 2D CNN

2.1. Dataset

The presented model begins by collecting image datasets from two distinct sources: the manufacturing factory village dataset and a Kaggle dataset, accessible at https://www.kaggle.com/. The Kaggle dataset comprises 2000 named characters and RGB film land of 14 crops, while the manufacturing factory village dataset is used in tandem with a Content-Based Image Retrieval (CBIR) framework to gather images, notably focusing on the product line brace image. Specifically, 13960 images depicting apple, blueberry, and cherry plant leaves are assembled for integration into the CBIR system.

2.2. Pre-processing

The initial preprocessing step involves preparing the images before model training. One of the techniques applied is resizing the pictures. The raw dataset is resized to dimensions of 256 by 256 pixels. This resizing helps standardize the images for further analysis.

Additionally, the resized plant flake images are utilized to address issues such as blur and noise. This is achieved by employing the middle channel, which evaluates each pixel against the median value of neighbouring pixels. Pixels exceeding this median value are considered, aiding in the reduction of salt and pepper noise while preserving important image details. The calculation of the median value is described by Equation 1.

$$Q\,(x, y) \;=\; \text{median}\,\{f\,(m, n)\} \tag{1}$$

Where (m, n) mx, y, the pixel intensity value within the kernel when m and n are equal is represented by the f (m, n) value.

2.3. Segmentation

Graph cut segmentation is a semi-automatic technique utilized for segmenting images into foreground and background elements. Unlike some other segmentation methods, graph cut segmentation does not rely heavily on precise initialization. Instead, it involves drawing lines, known as scribbles, on the image to indicate areas that belong to the foreground and those that belong to the background. This approach allows for a more interactive and intuitive segmentation process, where users can provide rough guidance to the algorithm without needing to delineate every region of interest precisely.

2.4. Feature extraction

Feature extraction is a crucial step in deep learning-based image processing tasks, where meaningful information is distilled from raw images to enable accurate analysis and decision-making. Three popular methods for feature extraction are 2D CNN, InceptionV3 and ResNet50.

2.4.1. 2D CNN

2D Convolutional Neural Networks are a class of deep learning models specifically designed for processing two-dimensional structured data, such as images. They have revolutionized various computer vision tasks, including image classification, object detection, segmentation and more. The architecture diagram of the projected 2D CNN is shown in Figure 2.

The convolution layer applies learnable filters (kernels) to input images to extract various features by sliding these filters across the input image. Pooling layers are often used in conjunction with convolutional layers to reduce the spatial dimensions of feature maps while retaining their essential information. Activation functions, such as ReLU (Rectified Linear Unit), are applied to the output of convolutional and pooling layers to introduce non-linearities into the network. The depth of a 2DCNN refers to the number of layers in the network, while the width refers to the number of filters within each layer. Deeper networks can capture more abstract and hierarchical features but may suffer from overfitting and vanishing gradients. Proper regularization techniques, such as dropout and batch normalization, are often employed to mitigate these issues.

2.4.2 InceptionV3

InceptionV3 has emerged as a pivotal convolutional neural network architecture renowned for its versatility and effectiveness in image classification tasks. Developed by Google researchers, it represents a significant advancement in deep learning for computer vision. Central to InceptionV3's success is its innovative Inception module, which employs parallel convolutional filters of varying sizes within a single layer, enabling the network to capture features at multiple scales simultaneously.

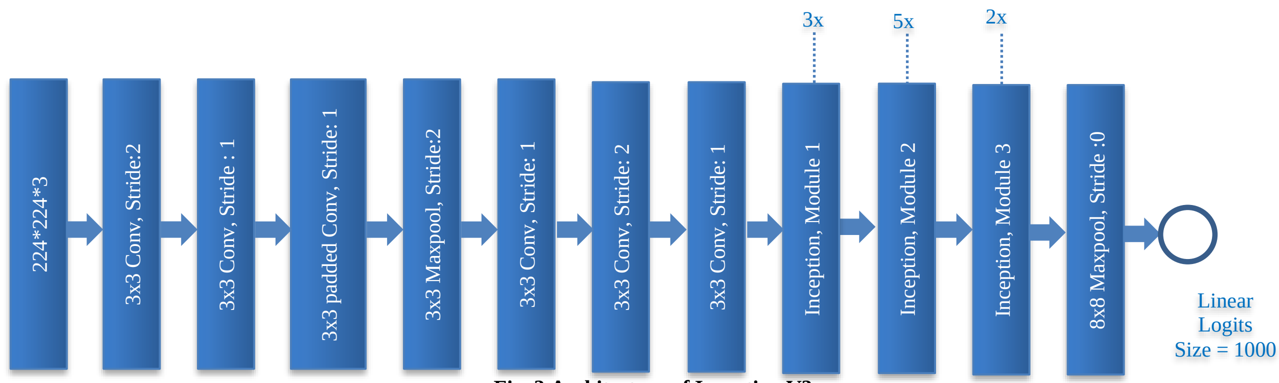

Fig. 3 Architecture of Inception V3

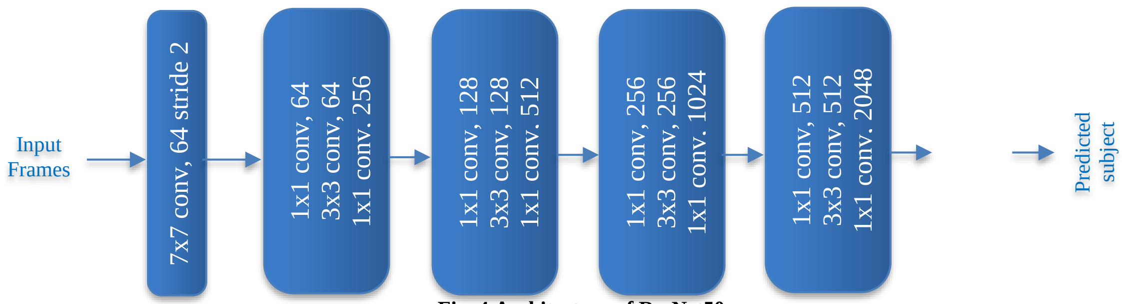

Fig. 4 Architecture of ResNet50

Additionally, InceptionV3 leverages auxiliary classifiers for regularization and batch normalization for stable training. Pre-trained InceptionV3 models, trained on large-scale datasets such as ImageNet which serve as powerful feature extractors for transfer learning tasks in image classification. The comparative analysis presented in this study underscores the efficacy and utility of InceptionV3 as a cornerstone in the realm of deep learning-based image classification. The architecture diagram of Inception V3 is depicted in Figure 3.

At the heart of InceptionV3 lies the Inception module, a cornerstone architectural element that enables the network to capture intricate and diverse features from input images. Unlike traditional convolutional layers that rely on fixed-size filters, the Inception module employs parallel convolutions with filter sizes ranging from 1x1 to 5x5. This multi-scale approach facilitates the extraction of features at various levels of abstraction by fostering robust image representation. In addition to the Inception module, InceptionV3 integrates auxiliary classifiers and batch normalization to enhance model generalization and stability during training. Auxiliary classifiers, strategically placed at intermediate layers, provide additional supervision signals by aiding in feature learning and regularization. Batch normalization, applied throughout the architecture, ensures stable training dynamics and accelerates convergence. Pre-trained InceptionV3 models, initialized with weights learned from large-scale datasets like ImageNet, serve as powerful tools for transfer learning in image classification tasks. By fine-tuning these pre-trained models on domain-specific datasets, practitioners can achieve state-of-the-art performance with minimal labelled data.

2.4.3. ResNet50

ResNet50, a variant of the ResNet (Residual Network) architecture, has emerged as a pivotal tool in deep learning for image classification tasks. By introducing skip connections that mitigate the vanishing gradient problem, ResNet50 revolutionized the field of deep learning by offering remarkable performance in image classification. Comprising 50 layers, ResNet50 is renowned for its ability to capture intricate details and patterns from images by making it highly effective in discerning between different classes. ResNet50 is characterized by its deep architecture, which is organized into blocks with residual units. The introduction of skip connections allows for the direct flow of information between layers by mitigating the challenges associated with training deep neural networks. This architectural innovation enables ResNet50 to effectively capture discriminative features from input images, which facilitates accurate classification. One of the primary strengths of ResNet50 lies in its feature extraction capabilities. The residual connections within the architecture enable the network to capture both low-level details and high-level semantic information from images. This enables ResNet50 to learn intricate patterns and subtle nuances inherent in different image classes, thereby enhancing its classification accuracy. The architecture diagram of ResNet50 is depicted in Figure 4.

ResNet50 can be trained using standard deep learning techniques named Stochastic Gradient Descent (SGD) with backpropagation. Transfer learning, wherein pre-trained ResNet50 models are fine-tuned on task-specific datasets, is a common approach to expedite training and improve performance in image classification tasks.

2.5. Feature Database

The Extracted features from each convolutional neural network architecture, namely 2DCNN, InceptionV3, and ResNet50, are meticulously organized within the feature database. Each image is associated with its respective set of extracted features.

Corresponding labels, representing the class or category to which each image belongs, are seamlessly integrated into the feature database. These labels provide crucial ground truth information essential for supervised learning tasks by enabling accurate classification and evaluation of model performance.

2.6. GBT Classifier

The GBT classifier is an ensemble learning technique that combines multiple decision trees in a sequence, where each subsequent tree corrects the errors made by the previous ones. It employs boosting by focusing more on misclassified data points with each iteration to reduce overall prediction error and utilizes gradient descent optimization to minimize a loss function by fitting new trees to gradient residuals. GBT models often incorporate regularization techniques to prevent overfitting, and they provide a measure of feature importance for understanding the contribution of each input feature to predictive performance. GBT classifiers are widely used in various domains due to their high predictive accuracy, flexibility and applicability to tasks with complex feature interactions.

The prediction of a GBT model is represented as the sum of predictions from each individual tree in the ensemble, as shown in Equation 2,

$$\check{y} = \sum_{i=1}^{N} f_i(x) \quad (2)$$

Where n is the number of trees in the ensemble. It determines the complexity of the model and the number of iterations during training. $f_i(x)$ is the prediction of the i^{th} decision tree in the ensemble for the input x. It represents the contribution of each tree to the final prediction.

$\check{y}$ is the final predicted output of the GBT model for the input x. It is the sum of predictions from all the decision trees in the ensemble. The y represents the true labels or target values associated with the input data x. It represents the ground truth against which predictions are compared during training.

GBT minimizes a loss function, typically represented as the sum of a differentiable loss function L over all training examples given in Equation 3.

$$\text{Loss} = \sum_{i=1}^{N} L(y_i, \check{y}_i) \quad (3)$$

The gradient of the loss function with respect to the predictions of the previous trees is calculated to guide the fitting of the next tree is derived as shown in Equation 4.

$$\text{Gradient} = -\frac{\partial L(y, \check{y})}{\partial \check{y}} \quad (4)$$

2.7. Similarity Measure

The similarity measure between the feature representation of a query image and the feature representations of images in the dataset is a crucial step in evaluating the performance of a CBIR system. The Euclidean distance between the feature vectors of the query image (F_q) and each dataset image (F_d) is computed as shown in Equation 5.

$$\text{Euclidean distance} = \sqrt{\sum_{i=1}^{n} (F_q[i] - F_d[i])^2} \quad (5)$$

Once the Euclidean distances are computed for each dataset image, the similarity between the query image and each dataset image will be assessed.

2.8. Performance Evaluation

Precision measures the proportion of retrieved images that are relevant to the query image. It is calculated as the ratio of relevant images retrieved to the total number of images retrieved, as shown in Equation 6.

$$\text{Precision} = \frac{\text{No of relevant images retrieved}}{\text{Total no of images retrieved}} * 100 \quad (6)$$

Recall measures the proportion of relevant images that are successfully retrieved out of all relevant images in the dataset. It is calculated as the ratio of relevant images retrieved to the total number of relevant images by using Equation 7.

$$\text{Recall} = \frac{\text{No of relevant images retrieved}}{\text{Total no of relevant images}} * 100 \quad (7)$$

Accuracy is a common metric used to evaluate the overall performance of a retrieval system, including CBIR. It measures the proportion of correctly retrieved images out of all images retrieved, regardless of their relevance to the query image. The accuracy is calculated as shown in Equation 8.

$$\text{Accuracy} = \frac{\text{No of images retrived}}{\text{Total no of images retrived}} * 100 \quad (8)$$

These metrics provide valuable insights into the effectiveness of the retrieval process that will be obtained in this study. High precision indicates that a large proportion of retrieved images are relevant, while high recall indicates that a large proportion of relevant images in the dataset are successfully retrieved. Balancing precision and recall is essential for achieving an optimal retrieval system that retrieves a high number of relevant images while minimizing irrelevant ones.

3. Result Analysis

A deep learning classifier is implemented to categorize plant leaves into eight distinct classes, including apple scab, cedar apple rust, apple black rot, apple healthy, blueberry healthy, blueberry rust, cherry healthy and cherry powdery mildew. The classifiers are trained using a dataset comprising 13,960 images encompassing three different types of plant leaves. Following the completion of the training phase, the images are shuffled within the database to assess the effectiveness of the Content-Based Image Retrieval (CBIR) system. The performance of the deep learning classifiers is evaluated using three key metrics: accuracy, precision and recall. These metrics provide insights into the classifiers' ability to accurately identify and classify plant leaves into their respective categories. A systematic approach is proposed to effectively categorize plant leaves based on their visual characteristics, aiding in the early detection and management of various diseases affecting apple, blueberry and cherry plants. For detailed information regarding the number of images used for training each class, refer to Table 1 provided in the document.

Table 1. The number of images used for eight classes

SL NO	Plant leaf classes	No of images
1	Apple scab	1750
2	Apple black rot	1655
3	Cedar apple rust	1855
4	Apple healthy	1800
5	Blueberry rust	1475
6	Blueberry healthy	1769
7	Cherry healthy	1756
8	Cherry powdery mildew	1900
Total		**13960**

The comparison is conducted utilizing deep learning algorithms, specifically leveraging 2DCNN, together with transfer learning models such as InceptionV3 and ResNet50. The evaluation involves comparing these algorithms in conjunction with various preprocessing and segmentation techniques. Preprocessing methods, including resizing, colour conversion (RGB to HSV) and median filtering, are applied to enhance the input images before feeding them into the models. Additionally, segmentation techniques utilizing superpixel graph cut methods are employed to partition images into meaningful regions by facilitating better feature extraction. For feature extraction, the selected algorithms, namely 2DCNN, InceptionV3 and ResNet50, are employed. These algorithms extract high-level features from the pre-processed and segmented images by enabling a more accurate representation of image content. The retrieval performance based on query image for the presented SBIR system is given in Table 2.

The image dataset consists of a collection of 13690 images with 8 different categories, including Each of these categories 1000 images. A set of 8 image queries has been applied to test the efficiency of our system and similar categories have been selected. The top 100 images are retrieved for each query image.

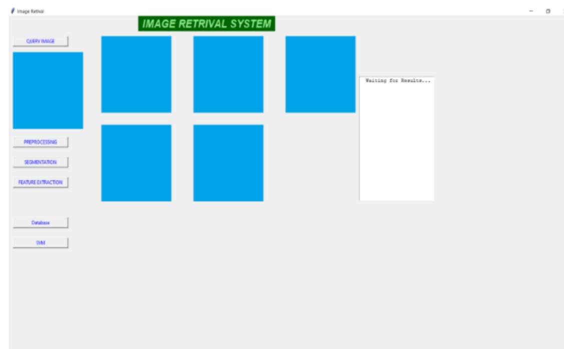

Fig. 5 GUI framework of CBIR system

The framework for plant disease identification utilizing a machine learning-based Content-Based Image Retrieval (CBIR) system is depicted in Figure 5. The retrieved sample images of eight plant leaf classes using this proposed CBIR method are presented in Figure. Comparative performance analysis of the proposed model in terms of accuracy, precision and recall is provided in Table 2.

By using the 2DCNN algorithm, Inception V3 algorithm and the Resnet algorithm, it will load the image into the GUI frame, which is where input the image and obtain the workings of the affair. Then, it will continue using the 2DCNN model. Upon inputting an image, the process will entail preprocessing, segmentation, feature extraction utilizing deep models, data training, CBIR Euclidean test and CBIR GB test. Subsequently, six distinct relevant images will be retrieved and displayed alongside the inputted image, as shown in Figure 6.

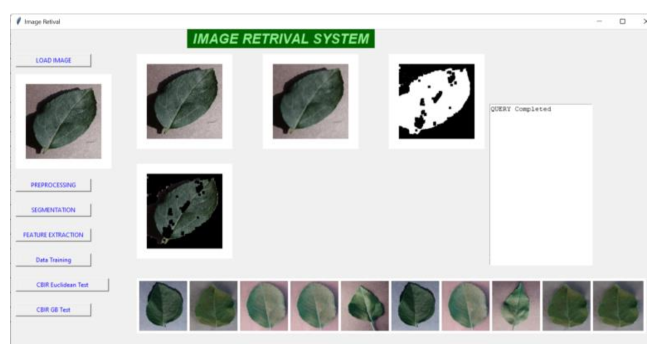

Fig. 6 GUI framework of CBIR system for 2D CNN

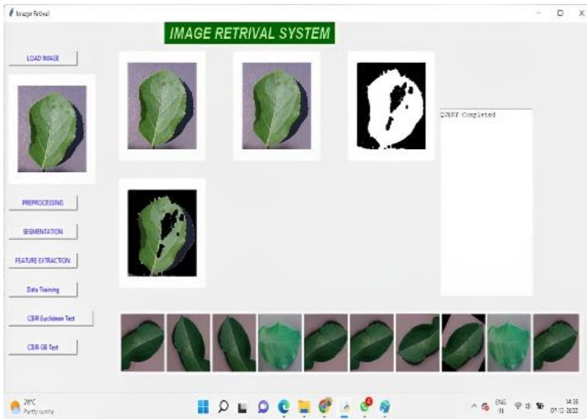

Fig. 7 GUI framework for inception V3

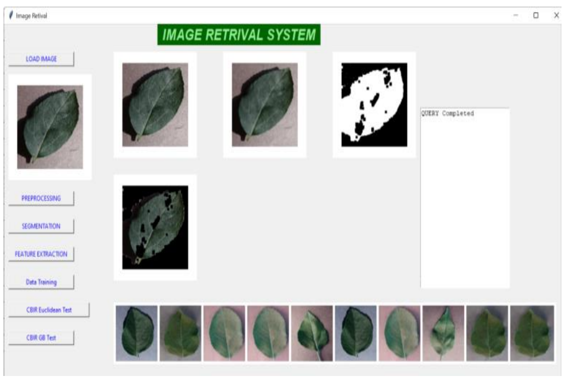

Fig. 8 GUI framework for Resnet50

Upon inputting an image, the procedure will involve employing the Inception V3 model. This encompasses preprocessing, segmentation, feature extraction utilizing deep models, data training, CBIR Euclidean test and CBIR GB test. Following this, seven distinct relevant images will be retrieved and exhibited alongside the inputted image, as shown in Figure 7.

When an image is inputted, the process involves utilizing the ResNet50 model. This includes preprocessing, segmentation, feature extraction using deep models, data training, the CBIR Euclidean test, and the CBIR GB test. Afterwards, eight distinct relevant images will be retrieved and displayed alongside the inputted image, as shown in Figure 8.

4. Conclusion

The proposed framework of a CBIR system integrates data collection, preprocessing, segmentation, feature extraction and similarity measures. The noise in the Kaggle factory splint dataset was effectively eliminated using standard filtered preprocessing techniques. Utilizing the superpixel graph cut method, the CBIR system inputs data into deep learning classifiers. The semantic point representation of the model was captured through 2D CNN, Inception V3 and ResNet50 on eight distinct types of factory leaf classes. A deep learning classifier assigns a prediction score based on the probability function of Euclidean distance and gradient boost to retrieve images similar to the query image. The performance of the CBIR model was assessed using the same delicate performance criteria. The delicateness of 2DCNN is 87.43 with Euclidean distance 86.52 with gradient boost. Inception V3 scores 94.43 with Euclidean distance 96.52 with gradient boost. For ResNet50, the scores are 97.31 with Euclidean distance and 98.29 with gradient boost.

Acknowledgment

We would like to thank Annamalai University for supporting this research. We would also like to thank our supporting staff from the Faculty of Engineering, Annamalai University, who provided insight and knowledge that considerably aided the research.

References

[1] Nishant Shelar et al., “Plant Disease Detection Using CNN,” *International Conference on Automation, Computing and Communication 2022 (ICACC-2022)*, India, vol. 44, pp. 1-6, 2022 [CrossRef] [Google Scholar] [Publisher Link]

[2] S. Ashwinkumar et al., “Automated Plant Leaf Disease Detection and Classification Using Optimal Mobile Net Based Convolutional Neural Networks,” *Materials Today: Proceedings*, vol. 51, pp. 480-487, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[3] Sk Mahmudul Hassan et al., “Identification of Plant-Leaf Diseases Using CNN and Transfer-Learning Approach,” *Electronics*, vol. 10, no.12, pp. 1-19, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[4] Gaganjot Kaur et al., “Review Analysis of Content-Based Image Retrieval for Plant Disease Identification Using Deep Learning Algorithms,” *AIP Conference Proceedings*, vol. 2555, no. 1, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[5] Youlian Zhu, and Cheng Huang, “An Improved Median Filtering Algorithm for Image Noise Reduction,” *Physics Procedia*, vol. 25, pp. 609-616, 2012. [CrossRef] [Google Scholar] [Publisher Link]

[6] Abdelhameed Ibrahim, and El-Sayed M. El-kenawy, “Image Segmentation Methods Based on Superpixel Techniques: A Survey,” *Journal of Computer Science and Information Systems*, vol. 1, no. 6, pp. 1-10, 2020. [Google Scholar]

[7] Thair A. Salih, Ahmed J. Ali, and Mohammed N. Ahmed, “Deep Learning Convolution Neural Network to Detect and Classify Tomato Plant Leaf Diseases,” *Open Access Library Journal*, vol. 7, no. 5, pp. 1-12, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[8] Yong Zhong, and Ming Zhao, “Research on Deep Learning in Apple Leaf Disease Recognition,” *Computers and Electronics in Agriculture*, vol. 168, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[9] R. Rani Saritha, Varghese Paul, and P. Ganesh Kumar, "Content Based Image Retrieval Using Deep Learning Process," *Cluster Computing*, vol. 22, no. 2, pp. 4187-4200, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[10] Nitin Arora, Aditya Kakde, and Subhash C. Sharma, "An Optimal Approach for Content-Based Image Retrieval Using Deep Learning On COVID-19 and Pneumonia X-Ray Images," *International Journal of System Assurance Engineering and Management*, vol. 14, no. 1, pp. 246-255, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[11] Arshiya Simran, P.S Shijin Kumar, and Srinivas Bachu, "Content Based Image Retrieval Using Deep Learning Convolutional Neural Network," *IOP Conference Series: Materials Science and Engineering*, vol. 1084, no. 1, pp. 1-6, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[12] Candice Bentéjac, Anna Csörgő, and Gonzalo Martínez-Muñoz, "A Comparative Analysis of Gradient Boosting Algorithms," *Artificial Intelligence Review*, vol. 54, no. 3, pp. 1937-1967, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[13] Shuhan Xia, Fuhanchen Zhang and Chengyi Zhang, "A Gradient Boosting Based Classification Technique for Assisted Prediction Algorithm Research," *In 2023 IEEE International Conference on Image Processing and Computer Applications (ICIPCA)*, Changchun, China, pp. 371-375, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[14] Sharada P. Mohanty, David P. Hughes and Marcel Salathé, "Using Deep Learning for Image-Based Plant Disease Detection," *Frontiers in Plant Science*, vol. 7, pp. 1-10, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[15] Srdjan Sladojevic et al., "Deep Neural Networks-Based Recognition of Plant Diseases by Leaf Image Classification," *Computational Intelligence and Neuroscience*, vol. 2016, no. 1, pp. 1-11, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[16] Mohammed Brahimi et al., *Deep Learning for Plant Diseases: Detection and Saliency Map Visualization*, Human–Computer Interaction Series, Springer, Cham, pp. 93-117, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[17] Artzai Picon et al., "Deep Convolutional Neural Networks for Mobile Capture Device-Based Crop Disease Classification in the Wild," *Computers and Electronics in Agriculture*, vol. 161, pp. 280-290, 2019. [CrossRef] [Google Scholar] [Publisher Link]